AIAA SciTech Forum
6-10 January 2020, Orlando, FL
AIAA Scitech 2020 Forum

10.2514/6.2020-1562

Petrov-Galerkin Projection-Based Model Reduction with an Optimized Test Space*

Gary Collins†, Krzysztof J. Fidkowski‡, and Carlos E. S. Cesnik§
Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109, USA

Computational modeling is a pillar of modern aerospace research and is increasingly becoming more important in the engineering process as computer technology and numerical methods grow more powerful and sophisticated. However, computational modeling remains expensive for many aerospace engineering problems, such as large-scale aeroservoelastic control problems and multidisciplinary design optimization. Model reduction methods have therefore garnered interest as an alternative means of preserving high-fidelity in computational fluid dynamics (CFD) at a much lower computational cost. Specifically, projection-based model reduction methods seek to identify highly-important modes of a full-order model (FOM), for use in a projection of the FOM to a low-rank but highly-accurate reduced model. Many projection-based model reduction techniques are based on proper-orthogonal decomposition (POD), in which the approximation/trial basis is obtained from a singular value decomposition of a set of solution samples of the full-order model. Many applications derived from POD use a Galerkin approach, in that the test space used to project the equations is the same as the trial space. Yet, research has shown accuracy and stability benefits of using a Petrov-Galerkin approach, in which the test and trial spaces are different. An active area of research concerns the determination of this test space. This paper adapts methods for defining test functions for discontinuous Petrov-Galkerin PDE systems to the case of reduced models. Following the derivation, a steady linear model reduction example is used to verify the mathematical framework presented. Following this, a steady nonlinear problem and an unsteady nonlinear problem are presented to test the robustness of the optimal test space and to compare the method's performance to the performance of other commonly-used POD methods.

I. Introduction

Advances in high-performance computing and numerical methods have enabled scientists and engineers to generate high-fidelity solutions to difficult computational fluids problems, such as high-degree of freedom unsteady simulations. Concurrently, the need for generating such high-fidelity solutions has increased with advances in computational design techniques such as multidisciplinary design optimization (MDO). This need has outpaced the advances in CFD modeling, and the ability to generate solutions from high-fidelity models is a bottleneck in modern aircraft design and optimization.

Model reduction, also known as reduced-order modeling, has garnered increasing attention as a means for alleviating these computational issues. Two general types of model reduction techniques that have gained interest within the aerospace community are interpolation-based and projection-based methods. Interpolation-based reduced-order modeling techniques attempt to create a surrogate model that directly maps inputs to outputs of a system. Advances in stochastic modeling and machine learning have increased the efficiency and accuracy with which interpolation-based reduced order models are able to predict outputs. However, a drawback of these techniques is the replacement of the physics of the system with a model that is often a "black-box."

*Manuscript for the 2020 AIAA SciTech CFD conference
†Ph.D. Candidate, Department of Aerospace Engineering, gggggggg@umich.edu
‡Associate Professor, Department of Aerospace Engineering, kfid@umich.edu, AIAA Senior Member
§Clarence L. "Kelly" Johnson Collegiate Professor of Aerospace Engineering, cesnik@umich.edu, AIAA Fellow

On the other hand, projection-based techniques project the high-fidelity system to a lower-dimensional space, thereby retaining some of the physics of the original equations. Many projection-based techniques are based on Proper Orthogonal Decomposition (POD), which has been used in aerodynamics since the late 1990s.[1–4] For POD-based models, the approximation of the state is performed with basis vectors that come from a singular value decomposition of a set of samples of the full-order model. There are two approaches for projection of the PDE system itself: Galerkin projection, where the test space equals the trial space, and Petrov-Galerkin projection, where the test space and trial space differ.

Galerkin POD (GPOD) is commonly used and has been shown to be successful in several engineering applications.[5–7] It is also the formulation for several other model reduction techniques, such as the continuous and the discrete empirical interpolation method[8,9] and the missing point estimation technique.[10] However, GPOD models can be susceptible to instability and inaccuracy. For systems where the Jacobian is symmetric, positive-definite, the stability of the FOM guarantees the stability of the GPOD ROM; however, this is not true if the Jacobian is not symmetric, positive definite, as is the case for many nonlinear systems of interest. This is because the GPOD formulation can have arbitrarily large residual values in the space orthogonal to the trial space.[11] Additionally, erroneous solutions can arise from under-resolved residuals, and thus the errors of GPOD models are difficult to assess.[12] Furthermore, the Galerkin projection of the residual and Jacobian of the system can suppress high-frequency characteristics of the flow, removing important small scale dissipative forces that exist in turbulent flows, and generally, GPOD model reduction is not seen as a viable approach for modeling systems with discontinuities or high amounts nonlinearity.

Because of the accuracy and stability issues present in pure GPOD, great interest in placed in Petrov-Galerkin POD (PGPOD) formulations. A common procedure is to form test basis vectors that transform the PGPOD problem to a residual minimization approach, where a solution of the reduced state is sought such as to minimize the L_2 norm of the full-order residual[11,13] (L_2POD). This method has been shown to be beneficial over GPOD for a variety of problems[14–16] and is an underlying framework for the Least-Squares Petrov-Galerkin method[16] and the Gauss-Newton with Approximated Tensors method.[17]

Drawing inspiration from the discontinous Petrov-Galerkin (DPG) work of Demkowicz and Gopalakrishnan[18,19] and the optimal test function DPG work of Kast and Fidkowski,[20,21] this paper introduces a different method for formulating test basis vectors for PGPOD systems that are designed to optimally predict the state. These optimal test basis vectors are formulated by solving an adjoint-like system, which transforms the residual error minimization problem to be equivalent to a least-squares state error minimization problem. To decrease the computational cost of computing the test basis vectors online, a reduced way of computing the optimal basis vector is derived. After deriving these test basis vectors, a scalar numerical example is presented to verify the PGPOD method, and this is followed by nonlinear steady and unsteady problems meant to demonstrate the strength of this method over GPOD and L_2POD.

II. Projection-Based Model Reduction

II.A. System Formulation and Projection

A system of ordinary differential equations, generally arising from spatially-discretized partial differential equations, can be written as,

$$G : \begin{cases} \boldsymbol{M}\frac{d\boldsymbol{x}}{dt} + \boldsymbol{R}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) = \boldsymbol{0} \\ \boldsymbol{y}(t) = \boldsymbol{y}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \end{cases} . \tag{1}$$

Generally, most fluid dynamics problems can be cast in this setting, with $\boldsymbol{x} \in \mathbb{R}^N$ a vector of flow state variables, $\boldsymbol{u} \in \mathbb{R}^p$ a vector of inputs (such as geometric parameters or boundary conditions), $\boldsymbol{y} \in \boldsymbol{R}^q$ a vector of outputs, and t the time. In the above system, $\boldsymbol{M} \in \mathbb{R}^{N\times N}$ is the (sparse) mass matrix, and $\boldsymbol{R} \in \mathbb{R}^N$ is the discretized spatial residual.

II.B. Proper Orthogonal Decomposition

Model reduction by means of projection reduces the degrees of freedom of the system by representing the state vector as a linear combination of basis vectors,

$$\boldsymbol{x} = \boldsymbol{V}\hat{\boldsymbol{x}}, \tag{2}$$

where $\boldsymbol{V} : \hat{\boldsymbol{x}} \rightarrow \boldsymbol{x} \in \mathbb{R}^{N \times n}$ is a matrix whose columns contain the basis vectors, and $\hat{\boldsymbol{x}}(t) \in \mathbb{R}^n$ with $n \ll N$ is the reduced state. Substituting (2) into (1), and dropping the time dependence notation for clarity, yields

$$\begin{cases} \boldsymbol{M} \frac{d}{dt} (\boldsymbol{V}\hat{\boldsymbol{x}}) + \boldsymbol{R}(\boldsymbol{V}\hat{\boldsymbol{x}}, \boldsymbol{u}, t) = \boldsymbol{0} \\ \boldsymbol{y} = \boldsymbol{y}(\boldsymbol{V}\hat{\boldsymbol{x}}, \boldsymbol{u}, t) \end{cases}.$$

The first equation above can be multiplied on the left with a left projection matrix, $\boldsymbol{W}^T$ to create the reduced system,

$$\begin{cases} \hat{\boldsymbol{M}} \frac{d}{dt} (\hat{\boldsymbol{x}}) + \boldsymbol{W}^T \boldsymbol{R}(\boldsymbol{V}\hat{\boldsymbol{x}}, \boldsymbol{u}, t) = \boldsymbol{0} \\ \boldsymbol{y} = \boldsymbol{y}(\boldsymbol{V}\hat{\boldsymbol{x}}, \boldsymbol{u}, t) \end{cases}, \tag{3}$$

where $\hat{\boldsymbol{M}} \equiv \boldsymbol{W}^T \boldsymbol{M} \boldsymbol{V}$. Note, it is assumed that both $\boldsymbol{V}$ and $\boldsymbol{W}$ are time invariant. (3) represents the reduced system, where the trajectory of $\hat{\boldsymbol{x}}$ is solved rather than $\boldsymbol{x}$, reducing the degrees of freedom from N to n.

Combining the temporal and spatial residual terms into a single term will simplify derivations, leaving

$$\bar{\boldsymbol{R}}(\boldsymbol{x}, \boldsymbol{u}, t) = \boldsymbol{0} \tag{4}$$
$$\tag{5}$$

and the reduced formulation reads

$$\hat{\bar{\boldsymbol{R}}}(\boldsymbol{V}\hat{\boldsymbol{x}}, \boldsymbol{u}, t) = \boldsymbol{0} \tag{6}$$
$$\tag{7}$$

with

$$\bar{\boldsymbol{R}} = \boldsymbol{M} \frac{d}{dt} (\boldsymbol{x}) + \boldsymbol{R}.$$
$$\hat{\bar{\boldsymbol{R}}} = \hat{\boldsymbol{M}} \frac{d}{dt} (\hat{\boldsymbol{x}}) + \boldsymbol{W}^T \boldsymbol{R}.$$

There are several aspects of the above system that categorize the model reduction method in use. A Galerkin projection framework formulates its system with, $\boldsymbol{W} = \boldsymbol{V}$, while a Petrov-Galerkin based framework uses $\boldsymbol{W} \neq \boldsymbol{V}$.

A common method for deriving the set of basis vectors $\boldsymbol{V}$ is through proper orthogonal decomposition, also known as principal component analysis, combined with the method of snapshots, introduced by Sirovich.[22] This method involves collecting a series of full-order solutions (called "snapshots"), performing a singular value decomposition of the set of snapshots, and truncating the resulting left singular vector set to create the state basis vector set. For this discussion denote the set of solution snapshots by

$$\boldsymbol{S} = [\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_K] \in \mathbb{R}^{N \times K},$$

where $\boldsymbol{s}_i \in \mathbb{R}^N$ is a solution for the full-order system for a given set of conditions and K is the number of snapshots collected.

For many discretizations, a singular value decomposition of the snapshot matrix suffices for computing the state basis vector set. However, for modal finite-element discretizations, such as the DG method used in this work, the discrete singular value decomposition will depend on the finite-element basis functions used in the discretization. To eliminate this dependence, a continuous spatial inner product is used, and the singular vectors and values are computed from an eigendecomposition of the corresponding normal matrix. This normal matrix takes the form,

$$\boldsymbol{K}_{\text{norm}} = \boldsymbol{S}^T \boldsymbol{M} \boldsymbol{S}, \tag{8}$$

where $\mathbf{M}$ is the mass matrix. The eigenvectors of this system represent the singular vectors of the snapshot set, and the corresponding eigenvalues are the squares of the corresponding singular values ($\sigma_i = \sqrt{\lambda_i(\boldsymbol{K}_{\text{norm}})}$).

Singular vectors have two important properties: they span the snapshot space, and each basis vector is associated with a singular value. The first property implies that if the snapshot set is a good representation of the solution space of interest, then so is the basis set. Obtaining a "good" representation of the solution space thus typically requires that the snapshots are obtained from effectively sampling the parameters and inputs of interest of the system. The second property effectively ranks each singular vector in order of importance to the recreation of the snapshot set. This means that basis vectors associated with relatively small singular values can be removed without very large losses in solution space approximation.

The left singular vectors are truncated, and this smaller set of basis vectors is used as the projection basis, $\boldsymbol{V}$. The criterion for which basis vectors to remove is usually the cumulative sum of the singular values compared to the total sum of singular values, where, for example,

$$E_\sigma = \frac{\sum_{i=1}^{n} \sigma_i^2}{\sum_{i=1}^{K} \sigma_i^2} \geq 99.9\% \tag{9}$$

is a typical threshold. Although this criterion measures the accuracy with which the state space can be represented, it does not guarentee that the actual system can be driven towards the desired solution within the state space.

II.C. Test Basis Construction for POD Systems

The following section derives the two different sets of Petrov-Galerkin POD test basis's. The first being the L_2POD test basis which is designed such that the reduced residual minimization is equivalent to the full-order residual minimization with the state constrained to the reduced space; and the second being an optimal set of test basis vectors constructed by making the residual error minimization problem equivalent to minimizing the least-squares error between the POD-ROM derived solution and the projected FOM derived solution.

Consider an unsteady, linear PDE,

$$\frac{\partial \boldsymbol{x}}{\partial t} + \sum_{i=1}^{d} \alpha_i \frac{\partial^i \boldsymbol{x}}{\partial \theta^i} = \boldsymbol{g}, \tag{10}$$

where d is the spatial order of the PDE, θ^i represents the i^{th} spatial dimension, α_i are constants describing the magnitude of the spatial derivatives, and $\boldsymbol{g}$ are the boundary conditions and internal constraints that describe the problem at hand. In a fully discretized form, the PDE can be transformed to a set of coupled linear equations,

$$c\boldsymbol{M}\boldsymbol{x} + \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{11}$$

where $\boldsymbol{M}$ is a temporal operator, c is a constant, $\boldsymbol{A}$ is the Jacobian of the system, and $\boldsymbol{b}$ contains the constraints of the continuous problem $\boldsymbol{g}$ and contributions from previous time steps. The choice of c depends on the time scheme used to solve the problem. For example, $c = 0$ for a steady problem, $c = \frac{1}{\Delta t}$ for an implicit Euler solution, and $c = \frac{3}{2\Delta t}$ for a BDF2 time stepping scheme is formed.

The unsteady residual of the FOM can be written as,

$$\bar{\boldsymbol{R}}(\boldsymbol{z}) = [c\boldsymbol{M} + \boldsymbol{A}]\,\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{0}, \tag{12}$$

which in reduced form is,

$$\hat{\bar{\boldsymbol{R}}}(\hat{\boldsymbol{x}}) = \boldsymbol{W}^T\,[c\boldsymbol{M} + \boldsymbol{A}]\,\boldsymbol{V}\hat{\boldsymbol{x}} - \boldsymbol{W}^T\boldsymbol{b} = \boldsymbol{0}. \tag{13}$$

II.C.1. L_2 Minimum Residual Test Basis Formulation

The minimum residual formulation can be defined in an L_2 sense with,

$$\hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{z}}} ||\boldsymbol{R}(\boldsymbol{V}\hat{\boldsymbol{z}})||_2^2 = \arg\min_{\hat{\boldsymbol{z}}} ||\,[c\boldsymbol{M} + \boldsymbol{A}]\,\boldsymbol{V}\hat{\boldsymbol{z}} - \boldsymbol{b}||_2^2, \tag{14}$$

which is equivalent to,

$$\hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{z}}} \left[\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\hat{\boldsymbol{z}}\right]^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\hat{\boldsymbol{z}} - 2\left[\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\hat{\boldsymbol{z}}\right]^T\boldsymbol{b}+\boldsymbol{b}^T\boldsymbol{b}. \tag{15}$$

This minimum can be found by setting the gradient of the above statement with respect to $\hat{\boldsymbol{x}}$ equal to zero. The resulting solution is,

$$\left[\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\right]^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\hat{\boldsymbol{x}} = \left[\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\hat{\boldsymbol{x}}\right]^T\boldsymbol{b}. \tag{16}$$

Thus, as stated in Bui Tanh *et al.*,[11] minimum residual problem is equivalent to a Petrov-Galerkin POD formulation with a test basis of $\boldsymbol{W} = \left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}$. This test function formulation has seen wide use within the reduced order modeling field, including as the foundation for several other methods (e.g. GNAT[23]).

II.C.2. Test Basis Construction for Optimal State Prediction

Let $\boldsymbol{x}_{\text{FOM}} \in \mathbb{R}^{N\times 1}$ be a solution derived from (12), and let $\hat{\boldsymbol{x}}_{\text{ROM}} \in \mathbb{R}^{n\times 1}$ be a solution derived from the reduced problem (13). Ultimately, the minimization of the error between $\boldsymbol{x}_{\text{FOM}}$ and $\hat{\boldsymbol{x}}_{\text{ROM}}$ is desired. Thus, rather than seeking a residual minimization, one can seek the state error minimization. If this minimization is defined in an L_2 sense, then the error of the POD-ROM solution is,

$$e_{\hat{\boldsymbol{x}}_{\text{ROM}}} = ||\boldsymbol{V}\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{x}_{\text{FOM}}||^2. \tag{17}$$

which is equal to,

$$e_{\hat{\boldsymbol{x}}_{\text{ROM}}} = \hat{\boldsymbol{x}}_{\text{ROM}}^T\boldsymbol{V}^T\boldsymbol{V}\hat{\boldsymbol{x}}_{\text{ROM}} - 2\hat{\boldsymbol{x}}_{\text{ROM}}^T\boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}} + \boldsymbol{x}_{\text{FOM}}^T\boldsymbol{x}_{\text{FOM}}, \tag{18}$$

Notice that $\boldsymbol{V}^T\boldsymbol{V}$ is the identity, as $\boldsymbol{V}$ is an orthonormal basis. The minimization of this error can be found by setting the gradient of the error with respect to $\hat{\boldsymbol{x}}_{\text{ROM}}$ equal to zero,

$$\frac{\partial e_{\hat{\boldsymbol{x}}_{\text{ROM}}}}{\partial \hat{\boldsymbol{x}}_{\text{ROM}}} = 2\hat{\boldsymbol{x}}_{\text{ROM}}^T - 2\boldsymbol{x}_{\text{FOM}}^T\boldsymbol{V} = \boldsymbol{0}^T, \tag{19}$$

$$\Rightarrow \left[\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right] = \boldsymbol{0}. \tag{20}$$

Thus, the closest that $\hat{\boldsymbol{x}}_{\text{ROM}}$ can be to $\boldsymbol{x}_{\text{FOM}}$, is the projection of $\boldsymbol{x}_{\text{FOM}}$ onto the space spanned by the vectors in $\boldsymbol{V}$. One can consider $\boldsymbol{V}\boldsymbol{x}_{\text{FOM}}$ to be the *ideal* solution ($\hat{\boldsymbol{x}}_{\text{ideal}}$) of a POD model. *Ideal* in the sense that it is the most faithful replication of the full-order model that is possible with the limited information contained in the basis $\boldsymbol{V}$. This is an unsurprising outcome, but it does allow one to consider formulating the POD-ROM in a way such that minimizing the residual of the reduced system would be equivalent to minimizing the error between $\hat{\boldsymbol{x}}_{\text{ROM}}$ and $\hat{\boldsymbol{x}}_{\text{ideal}}$.

If the goal of the model reduction is to satisfy (20) and (13) for $\hat{\boldsymbol{x}}_{\text{ROM}}$, $\boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}$ must also satisfy (13). Thus the error in the residual (13) when the input is $\hat{\boldsymbol{x}}_{\text{ROM}}$ and when the input is $\boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}$ must also equal zero.

$$\begin{aligned} e_{\hat{\bar{\boldsymbol{R}}}} &= \boldsymbol{W}^T\bar{\boldsymbol{R}}(\boldsymbol{V}\hat{\boldsymbol{x}}_{\text{ROM}}) - \boldsymbol{W}^T\bar{\boldsymbol{R}}(\boldsymbol{V}\boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}), \\ &= \left[\boldsymbol{W}^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{W}^T\boldsymbol{b}\right] - \left[\boldsymbol{W}^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}} - \boldsymbol{W}^T\boldsymbol{b}\right], \\ &= \boldsymbol{W}^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\left[\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right] \end{aligned} \tag{21}$$

If the test basis is chosen such that $\boldsymbol{W}^T = \left[\left[c\boldsymbol{M}+\boldsymbol{A}\right]^{-T}\boldsymbol{V}\right]^T$, then,

$$\begin{aligned} e_{\hat{\bar{\boldsymbol{R}}}} &= \left[\left[c\boldsymbol{M}+\boldsymbol{A}\right]^{-T}\boldsymbol{V}\right]^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\left(\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right), \\ &= \boldsymbol{V}^T\left[c\boldsymbol{M}+\boldsymbol{A}\right]^{-1}\left[c\boldsymbol{M}+\boldsymbol{A}\right]\boldsymbol{V}\left(\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right), \\ &= \boldsymbol{V}^T I_{N\times N}\boldsymbol{V}\left(\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right), \\ &= \boldsymbol{V}^T\boldsymbol{V}\left(\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right), \\ &= I_{n\times n}\left(\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right), \\ &= \left(\hat{\boldsymbol{x}}_{\text{ROM}} - \boldsymbol{V}^T\boldsymbol{x}_{\text{FOM}}\right) = \boldsymbol{0}. \end{aligned} \tag{22}$$

This choice for the test basis therefore makes the problem of finding $\hat{\boldsymbol{x}}_{\text{ROM}}$ where $e_{\hat{\bar{\boldsymbol{R}}}} = \hat{\bar{\boldsymbol{R}}} = \boldsymbol{0}$ equivalent to (20), and the optimal test basis for a linear POD system is found by solving,

$$[c\boldsymbol{M} + \boldsymbol{A}]^T \boldsymbol{W} = \boldsymbol{V}. \tag{23}$$

This definition of optimal test basis vectors is only exact for linear PDEs. For nonlinear PDEs, the following approximation is made,

$$\left.\frac{\partial \bar{\boldsymbol{R}}}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}^\eta}^T \boldsymbol{W} = \boldsymbol{V}, \tag{24}$$

where $\hat{\boldsymbol{x}}^\eta$ is the current value of $\hat{\boldsymbol{x}}$ in an iterative solver of the POD-ROM.

II.C.3. Reduced Form of Optimal Test Basis Vectors

Notice that if the Jacobian in (13) were to evolve with the residual minimization process, the optimal test basis vectors would need to be solved multiple times. In addition to this issue, solving for one optimal test basis vector in (23) requires a full-order linear solve. So in essence, the overall cost of solving a reduced model using full-order optimal test basis vectors is on the same order as the full-order problem, which is undesirable.

There are two ways to reduce this cost. First, temporarily freezing the Jacobian and the optimal test basis vectors will reduce the cost on the order of the length of freezing. However, this can become an issue if the solution's path to the *ideal* solution is made longer by the need for additional solver iterations and if perturbations off of the optimal path occur due to the freezing of the Jacobian.

Another way to reduce the cost of solving for the optimal test basis vectors is to formulate a reduced form of (23). This can be done by seeking a solution for $\boldsymbol{W}$ that is a linear combination of another set of basis vectors,

$$\boldsymbol{W} = \boldsymbol{\Psi}\boldsymbol{T}, \tag{25}$$

where $\boldsymbol{\Psi} \in \mathbb{R}^{N \times n_{\boldsymbol{\Psi}}}$ is the test search space and $\boldsymbol{T} \in \mathbb{R}^{n_{\boldsymbol{\Psi}} \times n}$ is a matrix of coefficients. Making the above substitution into (23) results in,

$$[c\boldsymbol{M} + \boldsymbol{A}]^T \boldsymbol{\Psi}\boldsymbol{T} = \boldsymbol{V}. \tag{26}$$

Assume that $n < n_{\boldsymbol{\Psi}} < N$, and that there is a finer set of basis vectors $\boldsymbol{V}_{\text{fine}}$, such that $\boldsymbol{V}_{\text{fine}} \in \mathbb{R}^{N \times n_{\boldsymbol{\Psi}}}$ and $\boldsymbol{V} \subset \boldsymbol{V}_{\text{fine}}$. Left multiplying the above equation by $\boldsymbol{V}_{\text{fine}}^T$ results in,

$$\begin{aligned}
\boldsymbol{V}_{\text{fine}}^T [c\boldsymbol{M} + \boldsymbol{A}]^T \boldsymbol{\Psi}\boldsymbol{T} &= \boldsymbol{V}_{\text{fine}}^T \boldsymbol{V}, \\
\boldsymbol{V}_{\text{fine}}^T [c\boldsymbol{M} + \boldsymbol{A}]^T \boldsymbol{\Psi}\boldsymbol{T} &= \mathbb{I}_{n_{\boldsymbol{\Psi}} \times n}, \\
\left[\boldsymbol{\Psi}^T [c\boldsymbol{M} + \boldsymbol{A}] \boldsymbol{V}_{\text{fine}}\right]^T \boldsymbol{T} &= \mathbb{I}_{n_{\boldsymbol{\Psi}} \times n},
\end{aligned} \tag{27}$$

where $\mathbb{I}_{n_{\boldsymbol{\Psi}} \times n}$ is a $n_{\boldsymbol{\Psi}}$ by n matrix composed of the identity matrix and a zero matrix,

$$\mathbb{I}_{n_{\boldsymbol{\Psi}} \times n} = \begin{bmatrix} \mathbb{I}_{n \times n} \\ \boldsymbol{0}_{(n_{\boldsymbol{\Psi}} - n) \times n} \end{bmatrix}.$$

(27) is the reduced form (rPGPOD) of (23). rPGPOD transforms the cost of computing the optimal test basis vectors from solving n linear problems of size N to solving n linear problems to size $n_{\boldsymbol{\Psi}}$. If $\boldsymbol{\Psi} = \boldsymbol{V}_{\text{fine}}$, then the following can be found,

$$\left[\boldsymbol{V}_{\text{fine}}^T [c\boldsymbol{M} + \boldsymbol{A}] \boldsymbol{V}_{\text{fine}}\right]^T \boldsymbol{T} = \mathbb{I}_{n_{\boldsymbol{\Psi}} \times n}, \tag{28}$$

and

$$\boldsymbol{W} = \boldsymbol{V}_{\text{fine}}\boldsymbol{T}. \tag{29}$$

The reduced formulation (28) seeks to find optimal test basis vectors for each trial basis vector as a linear combination of a finer set of trial basis vectors. This has some advantages. First, the cost of finding the optimal test basis vectors is reduced, as mentioned above. Second, $\boldsymbol{V}_{\text{fine}}$ can be produced in tandem with $\boldsymbol{V}$. However, a disadvantage will be that $\boldsymbol{W}$ will be restricted to exist within the fine projection space. If this space is not chosen appropriately losses in accuracy might occur. To see how these losses affect the accuracy of the model and to test the above theory, the following section presents a numerical example.

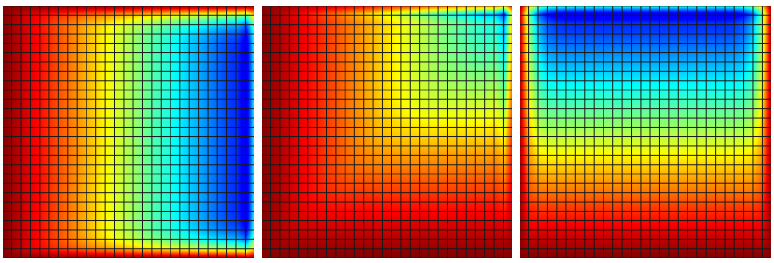
III. Scalar Model Example

A simple scalar advection-diffusion system within a square on a uniform grid serves as the linear test case and is a perfect problem for verifying the optimal PGPOD method. This is because it is expected that the solutions of the PGPOD model will converge to the projected FOM solutions due to the linear formulation of the advection-diffusion problem. The problem can be described with,

$$\nabla \cdot (\vec{v}x) - \nu \nabla^2 x + S = 0, \tag{30}$$

where $\vec{v}$ is a velocity field, x is the quantity of interest, ν is the diffusivity, and S is a source term.

The mesh for this case is constructed with 27^2 quadrilateral elements with bi-linear solution interpolation, resulting in 2916 degrees of freedom. The total lengths of the domain are 3. State snapshots are obtained by varying the flow direction, α, in the range $[0°, 90°]$ with a velocity magnitude of $||\boldsymbol{v}|| = 1$. The boundary conditions are all homogeneous Dirichlet (zero), and a constant unit source is added to the equation. The diffusivity is $\nu = 0.01$, making the unit length based Péclét number 100. The output of interest is flux of x integrated along the right boundary.



(a) $\alpha = 0°$ **(b)** $\alpha = 45°$ **(c)** $\alpha = 90°$

Figure 1: Solutions to the scalar test problem.

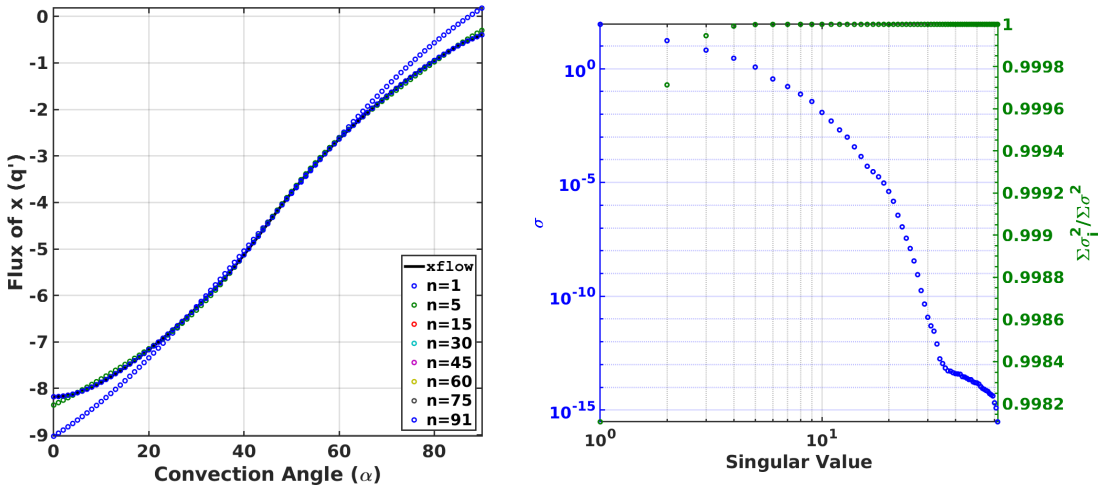
`xflow` – an in-house, high-order CFD solver[24] – is used to generate the FOM solutions for this numerical example and to compute the Jacobian and residuals for the ROMs. GMRES with line-Jacobi preconditioning is used to solve for the full-order optimal test basis vectors, while the rPGPOD test basis vectors are solved using a dense linear solver. The test basis vectors are solved vector-wise, and Gauss-Newton iterations is used to solve the POD-ROMs.

91 snapshots are used with even sampling of the internal flow angle. 99.9% of the total singular value energy is contained in the first 7 singular values. Figure 2 shows a plot of the singular values, the flux of x of the FOM, as well as the GPOD reconstruction.

As a small comparative study, the GPOD, L_2POD, and the full-order optimal PGPOD ROMs were solved with 5 basis vectors in the trial space, containing 99.7% of the total singular value energy. The results in Figure 3 demonstrate that the optimal PGPOD predictions and the *ideal* solutions align exactly, while the GPOD and the L_2POD models do not. This is true especially at the limits of the convection angles.

Taking a more comprehensive view, the errors of several GPOD, L_2POD, optimal PGPOD, and *ideal* solutions at a variety number of POD basis sizes are shown in Figure 4. With the exception of the POD basis sizes of 75 and 91, all of the optimal PGPOD models align with the *ideal* solutions exactly. Numerical error is suspected to have caused the rise in error of the *ideal* solutions at $n = 75, 91$. Additionally, for the coarser ROMs, the optimal PGPOD solutions are an order of magnitude more accurate than the GPOD solutions. However, the L_2POD model is fairly competitive with the PGPOD models.

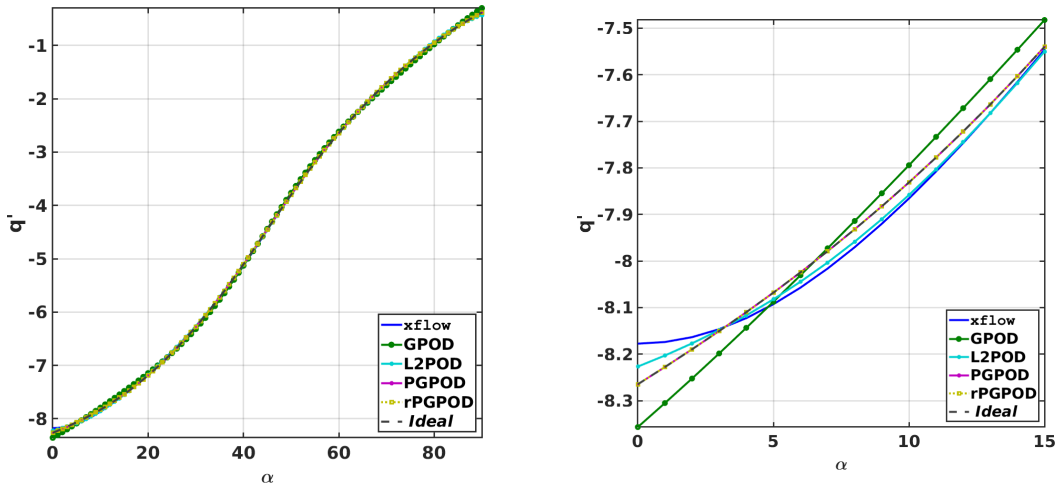
The results from the reduced optimal PGPOD model (rPGPOD) are also presented in Figures 3 and 4. The rPGPOD solutions are generated using all 91 basis vectors as the fine-space basis, and freezing of the



(a) Output of FOM and GPOD

(b) POD Singular Values

Figure 2: Scalar Test Problem: GPOD results and singular values.

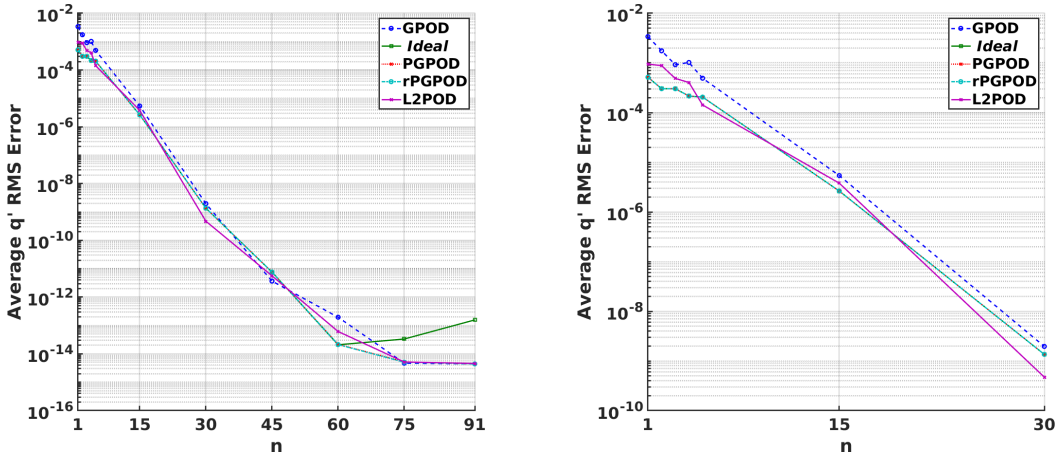


(a) $\alpha \in [0, 90]^\circ$

(b) $\alpha \in [0, 15]^\circ$

Figure 3: Scalar Test Problem: Comparisons between *Ideal*, GPOD, optimal PGPOD, reduced optimal PGPOD, and L_2POD reconstructions. The left image contains the entire parameter space, while the right image contains only the first 15 degrees of α in order to show how the PGPOD model and the *Ideal* solution remain (aligned even while incorrect) and how the GPOD and the L_2POD solution are misaligned.

Jacobian or the test basis vectors is not used. The rPGPOD solutions match the optimal PGPOD solutions and the *ideal* solutions.

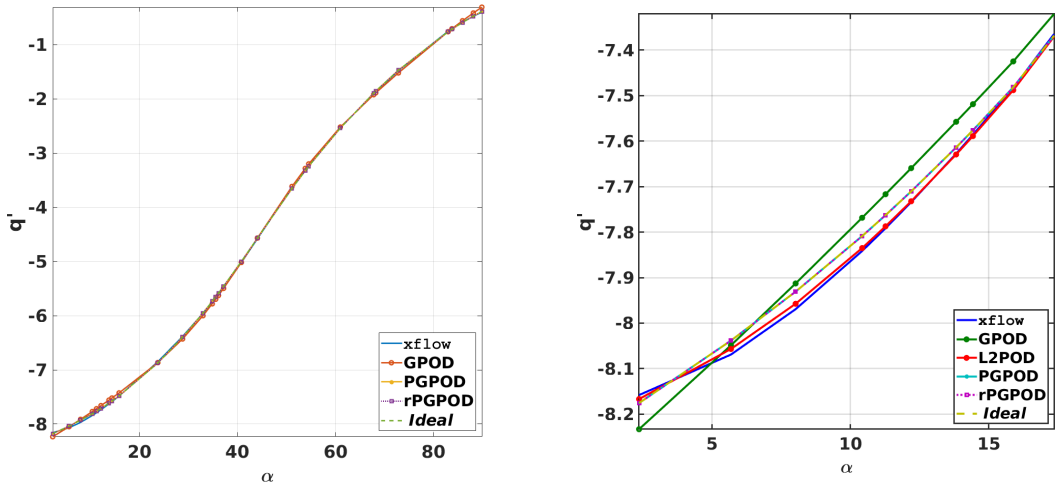


(a) $n \in [1, 91]$ **(b)** $n \in [1, 30]$

Figure 4: Scalar Test Problem: Errors of the *Ideal*, GPOD, optimal PGPOD, rPGPOD, and L_2POD reconstructions. The left image shows the errors trends for all of the POD basis sizes, n. The right image shows only the first 30 in order to highlight the improvements in accuracy of PGPOD over GPOD.

For at least reconstruction, the results of the full-order optimal PGPOD formulation comports with the earlier derived theory. Additionally, the information in the fine-space POD basis vectors is sufficient to produce optimal test basis vectors for the rPGPOD model. However, consider that all of the characteristics of the desired reconstructed solutions are already contained in the fine POD basis. To test the robustness of these approaches, predicting solutions for convection angles not already contained in the snapshot set is considered.

Thirty randomly-generated convection angles are chosen for ROM predictions. These convection angles are bounded by the limits of the angles in the snapshot set but are not contained in the snapshot set. The same POD-ROMs employed for snapshot reconstruction are used for prediction. The predictions from the POD-ROMs consisting of 5 basis vectors are shown in Figure 5. As before, the optimal PGPOD, rPGPOD, and the *ideal* solutions align exactly, while the GPOD and L_2POD solutions do not. The greatest errors in the GPOD solution occurs near the boundaries; however, the L_2POD solution aligns better with the output of the full-order model than the optimal POD models and the *ideal* solutions.



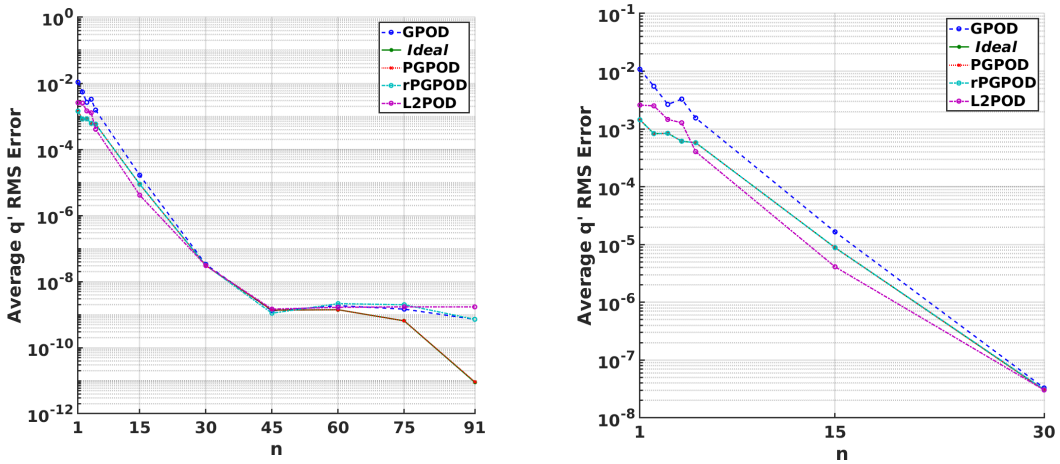
(a) $\alpha \in [0, 90]^\circ$ **(b)** $\alpha \in [0, 30]^\circ$

Figure 5: Scalar Test Problem: *Ideal*, GPOD, optimal PGPOD, reduced optimal PGPOD, and L_2POD predictions.

While the predicted solutions appear relatively similar to the reconstruction solutions, the errors with respect to the number of basis vectors, shown in Figure 6, are more revealing. For the coarser ROMs, the optimal PGPOD and the rPGPOD predictions match the *ideal* solution. However, divergence of the rPGPOD model is observed, beginning at about $n > 40$ as the trial space becomes finer. The rPGPOD predictions begin to follow the GPOD predictions, becoming exactly equal to the GPOD model at $n = 91$.

This clearly shows the limitations of the reduced formulation using a fine set of state basis vectors. The rPGPOD searches the fine projection space for the solution to the optimal test basis equation. Although the POD model increases in size, the fine-space projection space remains constant. With increases in the size of the trial space, the unique content of the test search space and the additional information that can be obtained by the test space decrease. Once the trial space and the test search space are identical, there is no additional information that the rPGPOD test space can obtain that is not already brought out by Galkerin projection.

Conversely, the full-order PGPOD model does not diverge and stays aligned with the *ideal* solution throughout the entire domain. This is because the full-order optimal PGPOD model is able to search the entire state space for the optimal test basis vectors. However, in most cases, it is not necessary to have a trial space size that approaches the size of the snapshot set, so these results still bode well for the rPGPOD as the coarser ROMs demonstrate the ability of the rPGPOD to produce good approximations of the optimal test space.



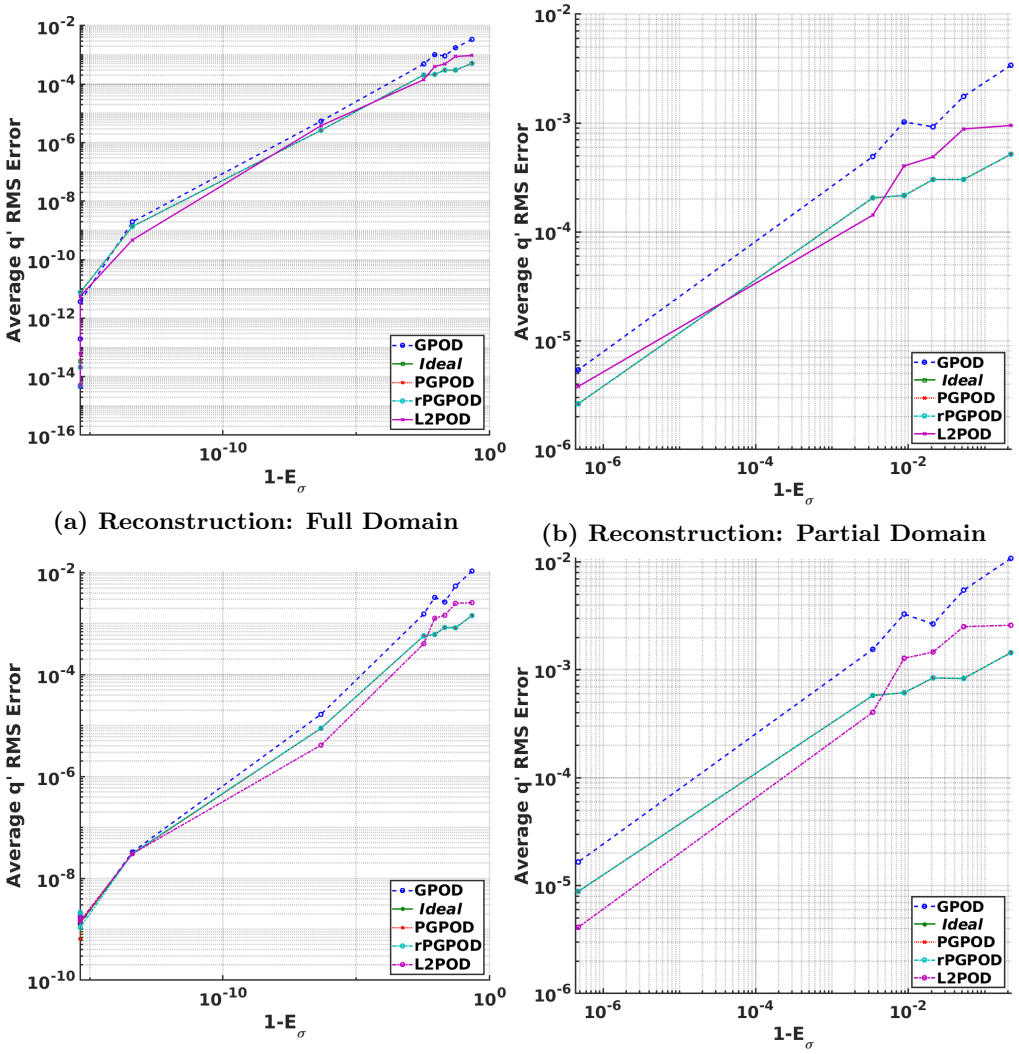
(a) $n \in [1, 91]$ **(b)** $n \in [1, 15]$

Figure 6: Scalar Test Problem: Errors of the *Ideal*, GPOD, optimal PGPOD, the rPGPOD, and L_2POD predictions.

While the error plots in Figures 4 and 6 reflect how the error behaves with respect to the number of POD basis vectors in the trial space, they lack information about the energy content of the POD model. This can be useful as the singular value energy (i.e. equation (9)) is typically used as a criterion for POD basis selection. By displaying the error of the system with $1 - E_\sigma$ as the x-axis, one can gauge how well the various POD models compare with an idea of which models would constitute a POD system that would be used in practice. As seen in Figure 7, the optimal POD models beat the GPOD model and the L_2POD, at least at the criterion set in Equation (9); however, this does not diminish the observation earlier that the L_2POD is at least competitive and sometimes superior to the optimal models.

These are remarkably good results for several reasons. First, the exactness with which the optimal PGPOD solutions match the *ideal* solutions is a verification of the earlier derived theory. Second, the rPGPOD solutions match the optimal PGPOD and the *ideal* solutions. This demonstrates that the reduced formulation of the optimal test basis vectors is sufficient for computing the test basis vectors for at least practical model sizes.

The performance of the GPOD model on this scalar example, while not matching the *ideal* solution, is still fair; additionally, the performance of the L_2POD model is not only competitive but also beats the optimal and ideal solutions at several instances. While it appears that the L_2POD beats the optimal PGPOD model for a scalar output, the flux on the right boundary is not fully-representative of the entire solution domain. State space domain errors are displayed in 8 for the prediction case with $n = 5$; additionally, the figure



(a) Reconstruction: Full Domain

(b) Reconstruction: Partial Domain

(c) Prediction: Full Domain

(d) Prediction: Partial Domain

Figure 7: Scalar Test Problem: Errors of the *Ideal*, GPOD, optimal PGPOD, the rPGPOD, and L_2POD reconstruction and prediction for linear problem with respect to $1 - E_\sigma$. Notice that the ROM size decreases moving left to right.

shows that the L_2POD model has the worst prediction of the entire domain of the system. Furthermore, the full-order optimal PGPOD, reduced optimal PGPOD, and *ideal* solution are all aligned and have the lowest error. It is observed that the *ideal* solution has the lowest error and the full-order optimal PGPOD aligns with the *ideal* solution for all of the tested problems; and where the reduced optimal PGPOD has sufficient test search space information, its domain solution also matches the *ideal* solution.

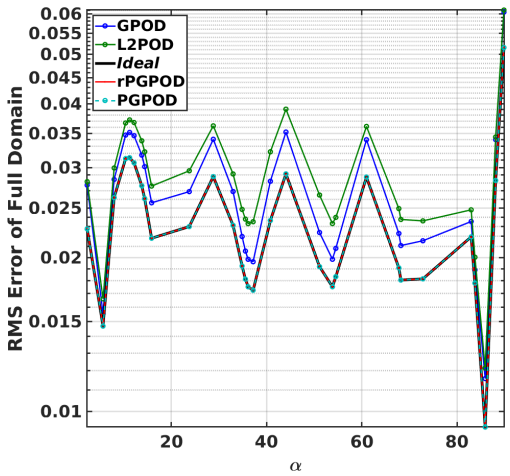


Figure 8: Scalar Test Problem: RMS error inside the entire domain for solutions generated with 5 basis vectors.

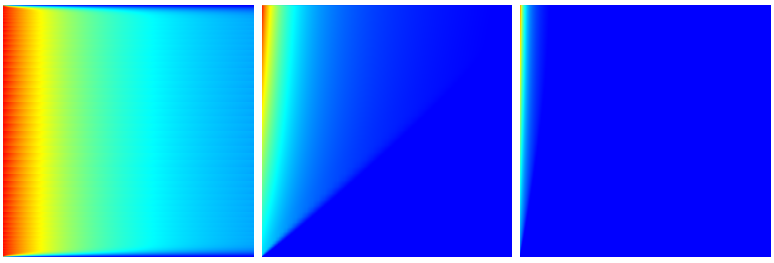
Although this example acts a good verification for the optimal PGPOD model, it does not test the robustness of the optimal PGPOD. As mentioned before, (23) and (28) are only approximations for nonlinear problems. It is expected that for nonlinear problems, the optimal PGPOD solutions may not align exactly with the *ideal* solutions; however, there may be more benefits to using the optimal PGPOD model over the GPOD and L_2POD model. This will be explored in the following section.

IV. Nonlinear Model Example

Building on the linear example above, nonlinearity is simulated by replacing the constant source term with one that is a nonlinear function of the state. The entire system can be written as,

$$\nabla \cdot (\vec{v}x) - \nu \nabla^2 x + S(x) = 0, \tag{31}$$

$$S(x) = S_0 x^2, \tag{32}$$



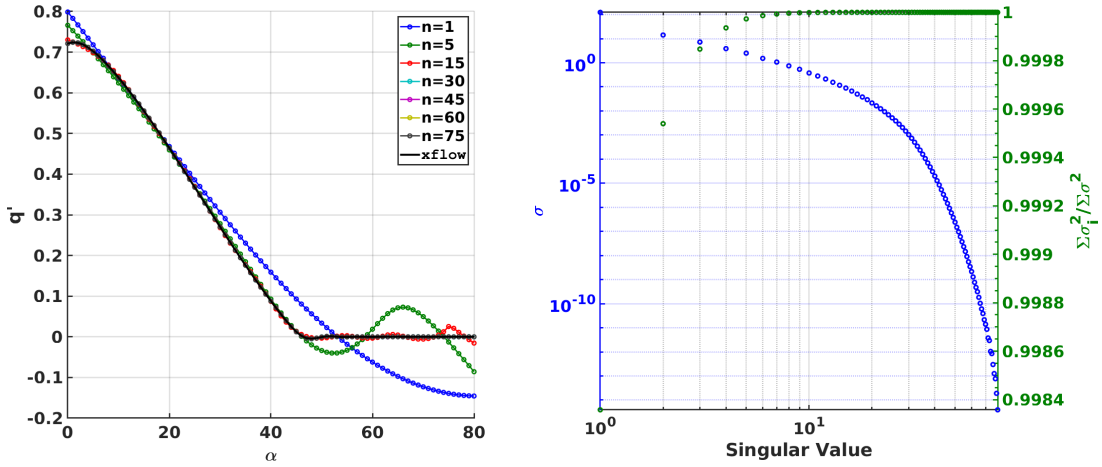
(a) $\alpha = 0^\circ$ **(b)** $\alpha = 40^\circ$ **(c)** $\alpha = 80^\circ$

Figure 9: Solutions to the nonlinear test problem.

with S_0 being a constant. For this example, the quadratic source coefficient is $S_0 = 1.0$, and the Péclét number is increased to 1000 through lowering the diffusion coefficient, ν, to 0.001. To maintain physical solutions, the mesh is refined such that 54 elements lies in the horizontal and vertical directions of the domain. In both directions, the length of each element decreases from the center, according to a cosine spacing. The element sizes are determined by $l_i = L\frac{1-\cos(\theta_i)}{2}$ where l_i is the width/height of element i numbered from the

top/left to bottom/right, θ_i is the corresponding angle from a uniformly spaced array $[0, \pi]$ rad, and L is the entire domain's width/height (in this case $L = 3$ for both width and height). α is sampled from 0° to 80° at 81 evenly distributed points. The GPOD, L_2POD, and reduced optimal PGPOD models are tested on both reconstruction of the original set of parameters and prediction of the midpoints between the sampled convection angles.

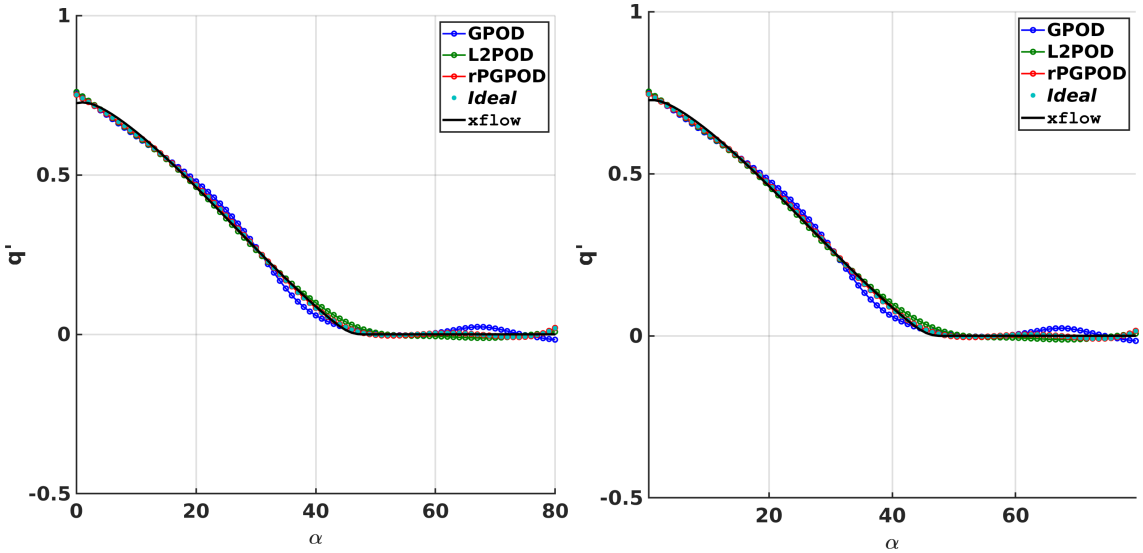
From Figure 9, we can tell that the additional nonlinear source term drastically changes the solution; additionally, as seen in Figure 9, the output changes drastically. Most importantly, notice that changes of α above approximately 50° have an insignificant effect on the flux of x.



(a) Output of FOM and GPOD **(b) POD Singular Values**

Figure 10: Nonlinear Test Problem: GPOD results and singular values.

Unsurprisingly, the GPOD model performs the worst of the three methods and is unable to fully resolve the right boundary. This can be seen in Figure 11, where the solution from the GPOD model with 5 basis vectors oscillates about the FOM model solution. Like before, the L_2POD model performs better, with a less pronounced oscillation, and the reduced optimal PGPOD model solution follows the *ideal* solution quite well.

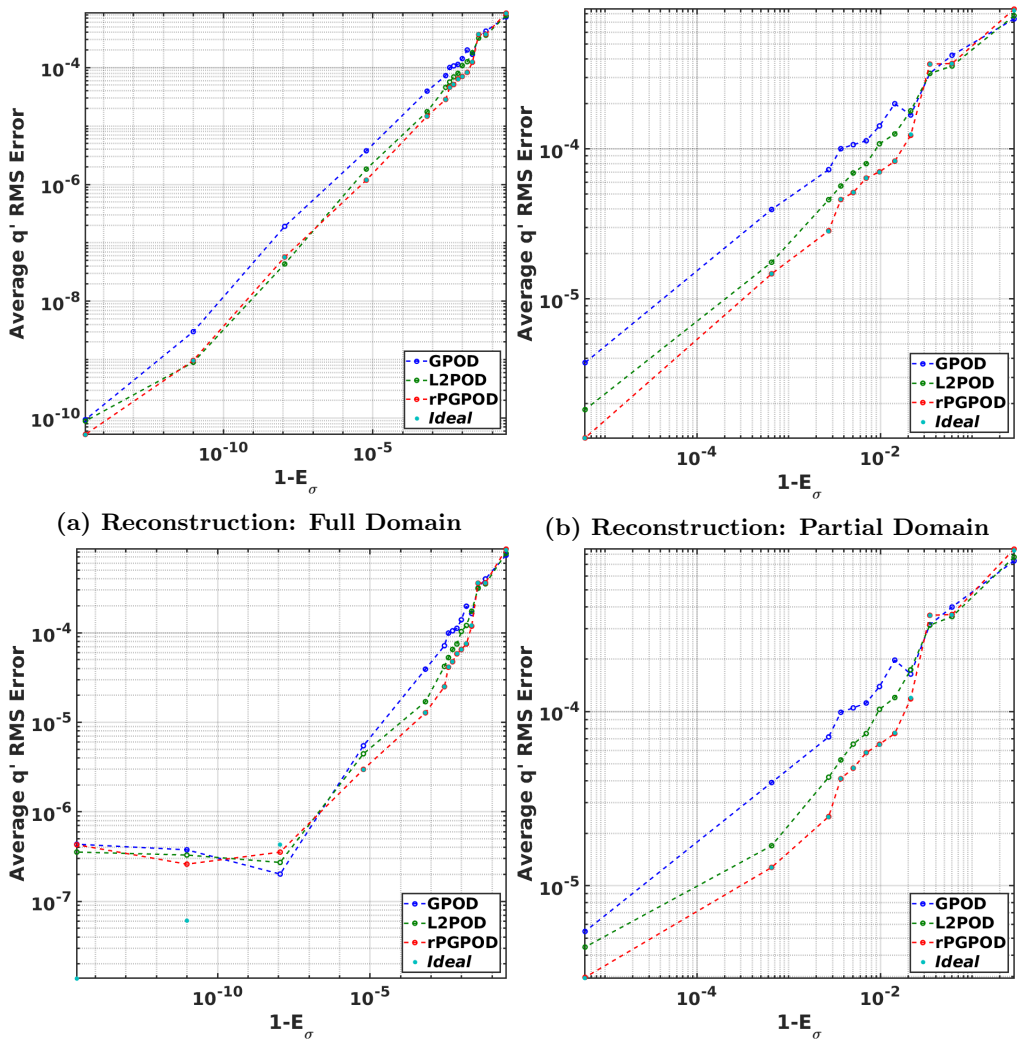


(a) Reconstructed Solutions **(b) Predicted Solutions**

Figure 11: Nonlinear Test Problem: Reconstruction and predicted results for $n = 5$ models.

The error of the ROMs here are slightly different than in the linear case, in a comparative sense. From Figure 12, there is a slight discrepancy between the *ideal* solution and the reduced optimal solution. This is even true for smaller ROMs, where a large amount of information can be introduced from the test search

space. Overall, in this nonlinear case, the reduced optimal PGPOD model performs better in comparison to the GPOD model and the L_2POD model than in the linear example in Section III. All of the reduced optimal PGPOD models performed better than the L_2POD and the GPOD models within range of singular value energies often used in an industrial setting ($99.9\% - 99.99\%$) in both prediction and reconstruction. These improvements are typically an order of magnitude greater than the GPOD model and 1-4 times greater than the L_2POD model. This is similar to the results seen in the previous section. Likewise, for prediction, the reduced optimal PGPOD model loses fidelity with the *ideal* solution as the trial space size approaches the test search space size, becoming equivalent to the GPOD model once the test search space equals the trial space.



(a) Reconstruction: Full Domain **(b) Reconstruction: Partial Domain**

(c) Prediction: Full Domain **(d) Prediction: Partial Domain**

Figure 12: Nonlinear Test Problem: Errors of the *Ideal*, GPOD, optimal PGPOD, the rPGPOD, and L_2POD reconstruction and prediction for nonlinear problem with respect to $1 - E_\sigma$. Notice that the ROM size decreases moving left to right.

Considering why the optimal PGPOD model performs better than the L_2POD model for the nonlinear problem than in the linear problem lies in the construction of the test space. One may think that the L_2POD model and the PGPOD models would have a similar outcome but from different mechanisms. The logic being that by minimizing the state reconstruction error, one should expect that the full-order residual would also be minimized; and, likewise, minimizing the full-order residual would also minimize the state reconstruction error. However, while the former is true, the latter is not necessarily true as locations of low residual error can be the source of high state error elsewhere, especially in highly-convective systems.

V. Unsteady Nonlinear Model Example

Finally, this section gives an example of the reduced optimal PGPOD model on an unsteady problem. The unsteady, nonlinear problem is built upon the steady, nonlinear problem, and can be described with,

$$\frac{\partial x}{\partial t} + \nabla \cdot (\vec{v}x) - \nu \nabla^2 x + S(x) = \mathbf{0}, \tag{33}$$

$$S(x) = S_0 x^2. \tag{34}$$

Unlike the previous two examples where constant Dirichlet boundary conditions are used, the boundary condition of the unsteady model is sinusoidal in time. The value of the scalar at the left boundary oscillates between the 0.0 and 1.0 at a frequency of 1.0 cycles/sec, while the value at the top and bottom of the domain oscillates between 0.0 and 2.0 at a frequency of 0.5 cycles/sec. The right boundary condition is kept constant at 0.0, a constant convection speed of 1.0 at $\alpha = 0^\circ$ is present, and the cosine mesh generated for the steady, nonlinear example is reused for this problem.

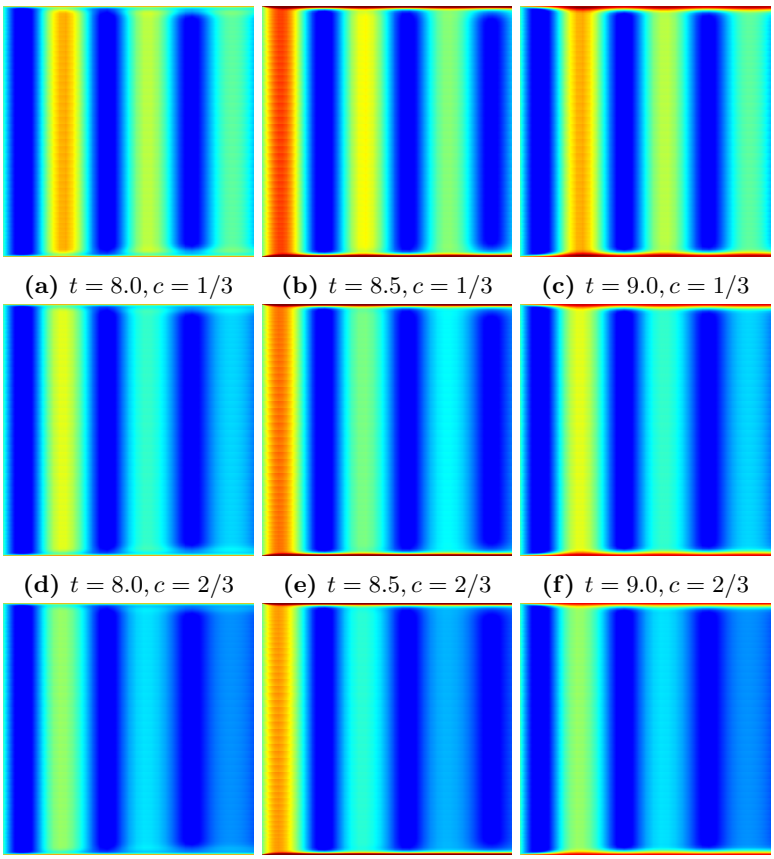
Three signals with different source constants are generated for creating the ROMs: $c = 1/3$, $c = 2/3$, and $c = 1$. The three signals are started from a steady state solution with the $t = 0.0$ conditions and simulated with BDF2 time-stepping from $t = 0.0$ to $t = 10.0$, with 1000 uniform time steps. The final 2 seconds of the simulations are placed into the snapshot pool. Two additional signals with different source constants are generated for testing the ROMs: $c = 1/2$ and $c = 5/6$. As seen in Figure 14 the peak amplitude of the flux of x at the right boundary varies nonlinearly with respect to the source constant, while the trough amplitude is nearly linear. These changes in amplitude can also be seen in the time slices contained in Figure 13. It can be concluded that the cause of the different regimes lies in the contribution of the source term to the system. At the peaks of the trajectories, the concentration of the state at the right boundary is relatively high, meaning that the source term, which contributes the nonlinearity to the system, is also high; inversely, at the troughs, the effect of the source term is much less pronounced as the concentration of the state at the right boundary is low.

As in the previous results, the singular value energy of the POD basis reaches a high proportion of the total energy with only a few basis vectors. 99% of the total singular value energy is captured with just the first 5 basis vectors; 99.9% with 9; 99.99% with 11; 99.999% with 18; and 99.9999% with 21. A comparison of these solutions can be seen in Figure 15, where BDF2 is used to march the reduced models through time and the rPGPOD models are solved with 100 state space basis vectors comprising the test search space. The first thing to note is that the *ideal* solution and the reduced optimal PGPOD model do not line up exactly. It is suspected that due to an accumulation of error through time, the two solutions should diverge. Clearly the predicted solutions corresponding to $c = 5/6$ is more difficult for the ROMs than the solutions corresponding to $c = 1/2$. One possibility is that the effect of the nonlinear term is greater for $c = 5/6$ than when $c = 1/2$, causing the linear assumptions made earlier to be less accurate.

For the coarsest ROMs, the $c = 5/6$ solutions contain a large amount of peak amplitude error as well as phase error for the GPOD solution. The $c = 1/2$ set of solutions are better; yet the trough error appears more substantial than for the $c = 5/6$ case. For $n = 9$, the ROMs should pass the criterion set by (9), and for the most part the ROMs perform fairly well. However, the L_2POD solution for the $c = 5/6$ contains large amounts of both amplitude and phase error. The same behavior is observed for the L_2POD solution with $n = 8$ and the GPOD model with $n = 7$. The remainder of the solutions fair well with the exception of the rPGPOD models with $n = 18$. Generating these solutions is considerably more difficult than the other rPGPOD solutions due to instability in the solution trajectory. For the test search space of 100 vectors, both rPGPOD test solutions diverged around the first peak amplitude. Decreasing the number of test search space vectors has the effect of allowing the system's trajectory to continue farther; however, divergence would still occur, typically around the peaks and trough of the system. Thus the number of test basis vectors is incrementally decreased until a rPGPOD solution is generated. This happens at 50 test search space basis vectors for $c = 1/2$ and 40 test search space basis vectors for $c = 5/6$. Unfortunately, these solutions are comparatively worse than the GPOD and L_2POD solutions with $n = 18$ system size. However in further testing of these two test cases, the rPGPOD solution does equal the GPOD solution when the test search space equals the trial space, similar to what is seen in the previous two sections.

The instability and inaccuracy due to an increase in basis vectors is a commonly observed issue with projection-based reduced order modeling, and perhaps the issues with accuracy and stability seen with the

rPGPOD model is related to the same phenomena. However, there are some remedies that can be employed to possibly resolve this issue. First, as noted before, the test search space can be constructed using basis vectors from sources other than the POD basis, such as a basis for adjoints, a basis for the residual, discrete Fourier vectors, etc. Each basis could be suited for different objectives, such as output prediction with the basis of adjoints and residual fidelity with the residual basis. Second, from ongoing work,[25,26] dual-weighted residual techniques can be used to rank vectors in a larger test search space in order of error of an output prediction. These ranks can then be used to construct a test search space for high-fidelity output prediction. These changes to the test search space may lead to greater stability and accuracy in the rPGPOD model for this problem.

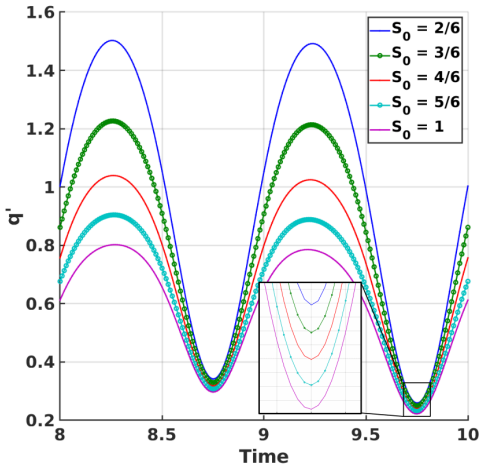


(a) $t = 8.0, c = 1/3$ **(b)** $t = 8.5, c = 1/3$ **(c)** $t = 9.0, c = 1/3$

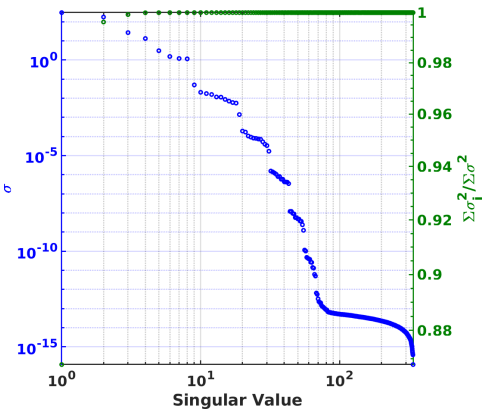
(d) $t = 8.0, c = 2/3$ **(e)** $t = 8.5, c = 2/3$ **(f)** $t = 9.0, c = 2/3$

(g) $t = 8.0, c = 1$ **(h)** $t = 8.5, c = 1$ **(i)** $t = 9.0, c = 1$

Figure 13: Solutions to the unsteady test problem.



(a) Solid lines → snapshots. Dotted lines → testing signals.



(b) POD Singular Values

Figure 14: Unsteady Test Problem: Training and testing signals and singular values.

VI. Conclusion

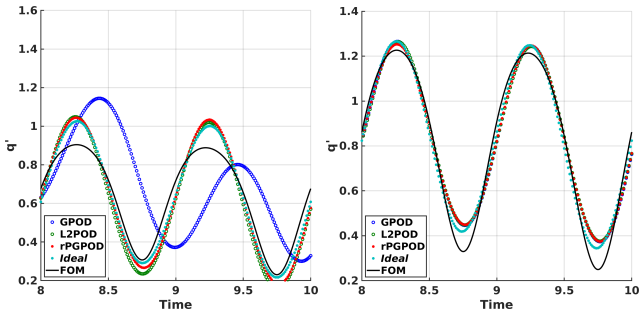
This manuscript introduces a method of constructing test basis vectors for a Petrov-Galerkin POD reduced-order model that optimally predicts the states. These test basis vectors are derived by making the residual error minimization problem equivalent to the L_2 minimization of the difference between the POD-ROM solution and the *ideal* solution, which is the projection of the FOM solution onto the trial space. An outcome of this way of defining the test basis is that the states will converge to the *ideal* solution of POD-ROM as the residual is driven to zero. In addition to a full-order model for the optimal test basis vectors, a reduced form is derived which seeks to find the test basis vectors as a linear combination of a finer set of trial basis vectors. Following these derivations, three numerical examples are given: a steady, linear case to verify the underlining theory; a steady, nonlinear case to test the strength of the method on nonlinear problems; and an unsteady, nonlinear case to see the robustness of the model on such problems. In each case an output of the system is predicted and used for evaluation.

Overall, for the linear case, the full-order optimal PGPOD formulation aligned perfectly with the *ideal* solution and the reduced formulation aligned perfectly for most POD sizes. Likewise, for the steady, nonlinear case, the reduced optimal PGPOD model stayed close to the *ideal* solution for most POD sizes. Finally, the reduced optimal PGPOD model performed fairly well on the unsteady, nonlinear problem but for at least one case instability of the model is present.

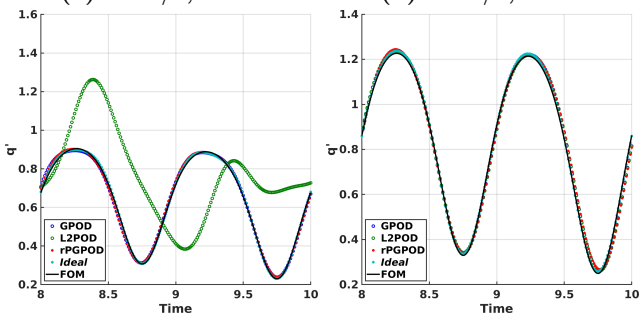
In addition to presenting optimal PGPOD models, comparisons between it and the Galerkin-POD model and the PGPOD model with an L_2 minimum residual test space are shown. For the steady, linear case, the reduced and full-order optimal PGPOD models performed overall better than the GPOD model with ROMs of the sizes that would be employed in industry (i.e. singular value energy between 99.9% and 99.999%); however the L_2POD model performance is competitive and occasionally beat the optimal PGPOD models. However, a greater benefit of the reduced optimal PGPOD model over the L_2POD model is observed for the nonlinear problem. This is due to the observation that state error minimization implies residual minimization whereas residual minimization does not imply state error minimization.

The optimal PGPOD models for the unsteady case are not as robust or accurate as in the steady cases. The reduced optimal PGPOD did not align perfectly with the *ideal* solution – likely due to the accumulation of error through time. Additionally, for a larger tested ROM size, the reduced PGPOD model suffered from inaccuracy and instability, behavior that is typically observed for projection based reduced order models. However, two possible remedies are suggested: changing the test search space in the reduced optimal PGPOD formulation and using dual-weighted residual techniques for test search space basis ranking and construction.

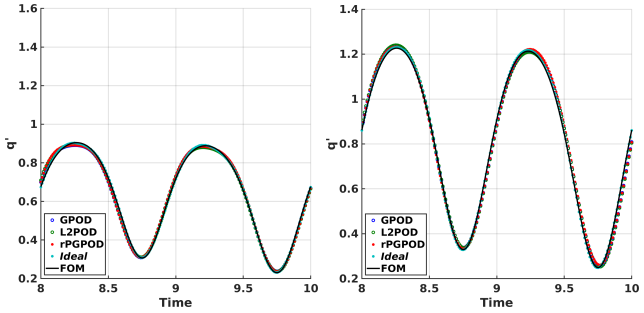
These remedies are planned for investigation in future work as well as the modifying the derivation of the optimal test basis vectors to optimally predict outputs by augmenting the desired error minimization. A similar method is employed by Kast *et al.*,[21] which inspired this work, to optimally choose test functions



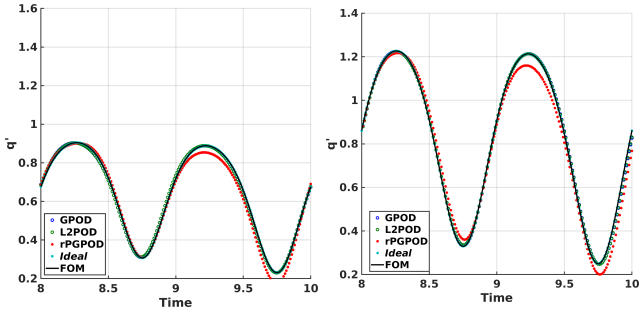
(a) $c = 5/6,\ n = 5$ **(b)** $c = 1/2,\ n = 5$



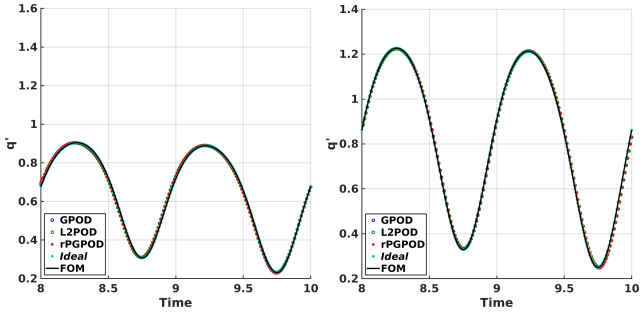
(c) $c = 5/6,\ n = 9$ **(d)** $c = 1/2,\ n = 9$



(e) $c = 5/6,\ n = 11$ **(f)** $c = 1/2,\ n = 11$



(g) $c = 5/6,\ n = 18$ **(h)** $c = 1/2,\ n = 18$



(i) $c = 5/6,\ n = 21$ **(j)** $c = 1/2,\ n = 21$

Figure 15: Unsteady Test Problem: Unsteady ROM solutions for various POD sizes.

for the discontinuous Petrov-Galkerin method.

Acknowledgments

The material is based on work supported by Airbus in the frame of the Airbus/Michigan Center for Aero-Servo-Elasticity of Very Flexible Aircraft. Special thanks to Hans Bleecke and Reik Thormann for their expert advice and review of this paper.

References

[1]Alonso, J. J., LeGresley, P. A., Legresley, P. A., and Alonso, J., "Airfoil design optimization using reduced order models based on proper orthogonal decomposition," 2000.

[2]Dowell, E., Hall, K., Thomas, J., Florea, R., Epureanu, B., and Heeg, J., "Reduced order models in unsteady aerodynamics," 1999.

[3]Willcox, K. and Peraire, J., "Balanced Model Reduction via the Proper Orthogonal Decomposition," *AIAA Journal*, Vol. 40, No. 11, Nov 2002, pp. 2323–2330.

[4]Rowley, C. W., "Model reduction for fluids, using balanced proper orthogonal decomposition," *Modeling And Computations In Dynamical Systems: In Commemoration of the 100th Anniversary of the Birth of John von Neumann*, World Scientific, 2006, pp. 301–317.

[5]Stabile, G., Hijazi, S., Mola, A., Lorenzi, S., and Rozza, G., "POD-Galerkin reduced order methods for CFD using Finite Volume Discretisation: vortex shedding around a circular cylinder," *Communications in Applied and Industrial Mathematics*, 2017.

[6]Lorenzi, S., Cammi, A., Luzzi, L., and Rozza, G., "POD-Galerkin method for finite volume approximation of Navier–Stokes and RANS equations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 311, 2016, pp. 151–179.

[7]Barone, M. F., Kalashnikova, I., Segalman, D. J., and Thornquist, H. K., "Stable Galerkin reduced order models for linearized compressible flow," *Journal of Computational Physics*, Vol. 228, No. 6, 2009, pp. 1932–1946.

[8]Galbally, D., *Nonlinear model reduction for uncertainty quantification in large-scale inverse problems: application to nonlinear convection-diffusion-reaction equation*, Ph.D. thesis, Massachusetts Institute of Technology, 2008.

[9]Chaturantabut, S., *Dimension reduction for unsteady nonlinear partial differential equations via empirical interpolation methods*, Ph.D. thesis, Rice University, 2009.

[10]Astrid, P., Weiland, S., Willcox, K., and Backx, T., "Missing point estimation in models described by proper orthogonal decomposition," *IEEE Transactions on Automatic Control*, Vol. 53, No. 10, 2008, pp. 2237–2251.

[11]Bui-Thanh, T., Willcox, K., and Ghattas, O., "Model reduction for large-scale systems with high-dimensional parametric input space," *SIAM Journal on Scientific Computing*, Vol. 30, No. 6, 2008, pp. 3270–3288.

[12]Lassila, T., Manzoni, A., Quarteroni, A., and Rozza, G., "Model order reduction in fluid dynamics: challenges and perspectives," *Reduced Order Methods for modeling and computational reduction*, Springer, 2014, pp. 235–273.

[13]Fang, F., Pain, C. C., Navon, I., Elsheikh, A., Du, J., and Xiao, D., "Non-linear Petrov–Galerkin methods for reduced order hyperbolic equations and discontinuous finite element methods," *Journal of Computational Physics*, Vol. 234, 2013, pp. 540–559.

[14]Li, J., Cai, J., and Qu, K., "Adjoint-based two-step optimization method using proper orthogonal decomposition and domain decomposition," *AIAA Journal*, Vol. 56, No. 3, 2018, pp. 1133–1145.

[15]He, J. and Durlofsky, L. J., "Constraint reduction procedures for reduced-order subsurface flow models based on POD–TPWL," *International Journal for Numerical Methods in Engineering*, Vol. 103, No. 1, 2015, pp. 1–30.

[16]Carlberg, K., Bou-Mosleh, C., and Farhat, C., "Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations," *International Journal for Numerical Methods in Engineering*, Vol. 86, No. 2, 2011, pp. 155–181.

[17]Carlberg, K., Cortial, J., Amsallem, D., Zahr, M., and Farhat, C., "The GNAT nonlinear model reduction method and its application to fluid dynamics problems," *6th AIAA Theoretical Fluid Mechanics Conference*, Vol. 2730, 2011, pp. 2011–3112.

[18]Demkowicz, L. and Gopalakrishnan, J., "A class of discontinuous Petrov–Galerkin methods. Part I: The transport equation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 199, No. 23-24, 2010, pp. 1558–1572.

[19]Demkowicz, L. and Gopalakrishnan, J., "A class of discontinuous Petrov–Galerkin methods. II. Optimal test functions," *Numerical Methods for Partial Differential Equations*, Vol. 27, No. 1, 2011, pp. 70–105.

[20]Kast, S. M., *Methods for Optimal Output Prediction in Computational Fluid Dynamics.*, Ph.D. thesis, University of Michigan, Department of Aerospace Engineering, 2016.

[21]Kast, S. M., Dahm, J. P., and Fidkowski, K. J., "Optimal test functions for boundary accuracy in discontinuous finite element methods," *Journal of Computational Physics*, Vol. 298, 2015, pp. 360–386.

[22]Sirovich, L., "Method of snapshots," *Quarterly of applied mathematical*, Vol. 45, No. 3, 1987, pp. 561–571.

[23]Carlberg, K., Farhat, C., Cortial, J., and Amsallem, D., "The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows," *Journal of Computational Physics*, Vol. 242, 2013, pp. 623–647.

[24]Fidkowski, K. J. and Roe, P. L., "An entropy adjoint approach to mesh refinement," *SIAM Journal on Scientific Computing*, Vol. 32, No. 3, 2010, pp. 1261–1287.

[25]Meyer, M. and Matthies, H. G., "Efficient model reduction in non-linear dynamics using the Karhunen-Loève expansion and dual-weighted-residual methods," *Computational Mechanics*, Vol. 31, No. 1, May 2003, pp. 179–191.

[26]Collins, G., Fidkowski, K., and Cesnik, C. E. S., "Output Error Estimation for Projection-Based Reduced Models," *AIAA Aviation 2019 Forum*, 2019, p. 3528.